

or H.265. Such information could be fed back to the filter optimization stage to further refine the downsampling process. For example, if it is known that the encoding would be at a particular bitrate and chroma precision below a certain level will be lost due to quantization, that could be considered during the filtering process to either reduce complexity or improve quality. Noise characteristics of the signal, as well as the denoising impact of the filters and/or of the encoding process, may also be considered for the optimization and the filtering process.

[0079] It should be noted that all the operations above could be applied not only on non-constant luminance Y'CbCr data, but also constant luminance Y'CbCr data, or other color spaces such as Y'u'v' (CIE 1976), Y'u"v", the IPT color space, or other color spaces that may be appropriate for video encoding.

[0080] Chroma Re-Sampling

[0081] Conventional color transform techniques use a non-linear transfer function (the transfer function or transfer domain). This is done in an effort to perform an initial basic compression of the signal to a fixed bit depth using some subjectively defined process (the transfer function). However, operations performed on color components in the transfer domain may result in a loss of precision but also errors, given the non-linearity of the transform function. These could be remedied by performing operations in the linear domain instead. The process may also be followed by additional color format conversion steps that aims at decorrelating color signals and isolating, commonly, intensity/brightness information from colour.

[0082] FIG. 7 is an example dataflow diagram for converting R'G'B' to Y'CrCb. To perform such operations, Y' in the non-constant luminance case may be calculated at block **702** as above, i.e., $Y' = (M[1,1]*R' + M[1,2]*G' + M[1,3]*B')$. Then the Y_ncl, B, R, and G components (in the linear domain) may be calculated in blocks **704a-704d** using the inverse transform function represented as $ITF()$. For example, Y_ncl can be optionally computed as $Y_{ncl} = ITF(Y')$, and the other components could be computed as $B = ITF(B')$, $R = ITF(R')$, and $G = ITF(G')$. Then the downsampled version of each component may be calculated at blocks **706a-706d** by applying a downsampling filter $Down()$ to the linear components. Then the non-linear version of each downsampled component may be calculated using the transfer function $TF()$ at blocks **707a-707d**. Using the downsampled, non-linear version of the color components (B', R', and G'), the Y, Cb, and Cr components in the transfer domain may be calculated (as above) at blocks **710**, **712**, and **714**, respectively. Alternatively, instead of the downsampled G' data we can use the downsampled Y_ncl data to generate the Cb and Cr components (as above). It should be noted that that derivation would not be the same as using the downsampled G' data, since the downsampled Y_ncl is derived directly from Y', which involves the non-linearities of the transfer function. In this scenario, an appropriate decoder that also considers the downsampled Y_ncl is necessary in deriving the final signals.

[0083] According to an embodiment, the calculation of the G component may not be necessary. The Y' value calculated initially at the Y conversion block from the R', G', and B' components may be output as the Y' component.

[0084] According to an embodiment, the Y' component after stage **704b** may not be available. For example, Y2' may

not be generated directly from the downsampled RGB values. Then Cr and Cb may be generated from R2, B2, and G2.

[0085] In an embodiment, we may determine that it is best to use a linear method for generating the downsampled version for one component, and a non-linear for the other. It is also possible to select that, for some instances, the non-linear method would be used for both, or a combination of the two could be considered. This decision could be made adaptively or based on some predefined conditions. For example, this can be determined by evaluating for both cases the distortion introduced in each scenario, as in the adaptive filtering process, compared to the original Cb and Cr components (without downsampling). An alternate color space could also be considered instead for this evaluation (e.g. RGB or XYZ), given an appropriate distortion measurement as discussed earlier. All possible or a subset of the combinations could be considered, e.g., consideration of non-linear downsampled Cb with linear downsampled Cr or/and vice versa, linear both, non-linear both, etc. The decision process could be done for an entire image, region of an image, or even a single sample. Additional combinations could be generated by averaging the samples together from each of the conversion processes. Averaging could be a simple average or weighted average based on some characteristics of the signal, or based on distortion performance.

[0086] The process of downsampling in the linear domain or the adaptive technique may be implemented in both the constant luminance and non-constant luminance domain, given the characteristics of each particular conversion, and may also be applied in alternative color spaces, such as Y'u'v', Y'u"v", IPT or others. In those cases, for example, conversion to an intermediate color space other than RGB may be required. In particular, conversion to XYZ or other color spaces such as LMS, e.g., for the IPT case, may first be required. Then downsampling is performed in those spaces to generate the downsampled versions, and the appropriate downsampled signals that correspond to the color components are derived from these images.

[0087] Closed Loop Conversions

[0088] Conventionally, each color component of image data is quantized and converted independently of each other. For example, the Y', Cb, and Cr components, in the constant luminance or non-constant luminance domain, may be calculated and then each will be quantized by applying a quantizing function $Q()$. However, once the Yq component is calculated, this value may be used to generate the Cb/Cr components. The same consideration could be made for other color spaces where there might be a dependency between the color components.

[0089] FIG. 8 is an example dataflow diagram for converting R'G'B' to YCbCr. In this example, in the non-constant luminance domain, upon receiving the RGB input, Y' may be calculated (in the transfer domain) at block **802** as above ($Y' = M[1,1]*R' + M[1,2]*G' + M[1,3]*B'$). Then the quantized value of Y' may be calculated at block **804** and, using the quantized value, Yr' may be reconstructed by inverse quantization at block **806**. Then, using the reconstructed value of Yr', Cb' and Cr' may be calculated as described above ($Cb' = (B' - Yr') * Sb$ at block **810** and $Cr' = (R' - Yr') * Sr$ at block **808**). The quantized values of Cr and Cb may then be calculated using a quantization function at blocks **812** and **814**, respectively. The downsampling process, such as in the case of converting the data to 4:2:2 or